UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/770,093	02/02/2004	Daniel J. Miller	MS1-631USC1	9988

22801 7590 06/26/2008

LEE & HAYES PLLC
421 W RIVERSIDE AVENUE SUITE 500
SPOKANE, WA 99201

EXAMINER	
PATEL, HARESH N	

ART UNIT	PAPER NUMBER
2154	

MAIL DATE	DELIVERY MODE
06/26/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/770,093	MILLER ET AL.	
	Examiner	Art Unit	
	Haresh N. Patel	2154	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *17 March 2008*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-20* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-10,12-17,19 and 20* is/are rejected.

7)☒ Claim(s) *4,11 and 18* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *02 February 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *See Continuation Sheet*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

Continuation of Attachment(s) 3). Information Disclosure Statement(s) (PTO/SB/08), Paper No(s)/Mail Date :2/2/2004, 4/13/2004, 5/10/2004, 7/15/2004, 12/20/2004, 5/2/2005, 12/7/2005, 12/7/2005(2), 3/8/2006, 8/30/2006, 11/15/2006, 04/13/2007, 07/19/2007, 11/02/2007, 01/24/2008, 06/02/2008.

DETAILED ACTION

1. Claims 1-20 are subject to examination. Claims 4, 11 and 18 are allowable but objected to.

Priority

2. Applicant's claim for domestic priority, as claimed on oath of this application, under 35 U.S.C. 119(e) is acknowledged.

Terminal Disclaimer

3. The terminal disclaimer filed on 3/17/2008 disclaiming the terminal portion of any patent granted on this application which would extend beyond the expiration date of 6,983,466 patent has been reviewed and is accepted. The terminal disclaimer has been recorded.

Specification

4. The disclosure is objected. Some of the informalities are: page 7 of the specification is missing description of figure 37. The status of the copending applications at pages 1, 8-10, etc., needs to be updated (including replacement of application numbers and/or attorney docket numbers with the patent numbers). Please refer to the application 09/731,490. Appropriate correction is required.

Drawings

5. The figures submitted on the filing date of this application are acknowledged.

Information Disclosure Statement

6. An initialed and dated copy of the applicant's IDS form 1449, is attached to the instant Office action, please see attachments section of the attached form PTO-326 containing paper dates.

Claim Rejections - 35 USC § 102

7. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.
> (a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

8. Claims 1-3, 5-10, 12-17, 19, 20 are rejected under 35 U.S.C. 102(e) as being anticipated by Parry et al. 6,535,920 (Hereinafter Parry).

9. Referring to claim 1, Parry discloses a method of generating a filter graph for a user-defined processing project (e.g., col., 3), the method comprising: analyzing the project for multiple accesses to a single source of media content (e.g., col., 6); determining that the multiple accesses cannot be combined and/or share a common processing chain (e.g., col., 6); and coupling a single instance of the media source to the one or more processing chains through a software object to satisfy the multiple accesses without invoking a commensurate number of multiple instances of the media source (e.g., col., 6).

10. Referring to claim 2, Parry discloses the claimed limitations as rejected above. Parry also discloses receiving a request for content at the software object; and issuing a seek command from the software object to the media source to retrieve the media for presentation to a requesting processing chain (e.g., col., 7).

11. Referring to claim 3, Parry discloses the claimed limitations as rejected above. Parry also discloses wherein the method is implemented by a render engine, exposed from an operating system to a media processing system executing on a computing system (e.g., col., 6).

12. Referring to claim 5, Parry discloses the claimed limitations as rejected above. Parry also discloses identifying multiple simultaneous access to the media source; and invoking a commensurate number of software objects, coupling a commensurate number of instances of the media source to processing chains to satisfy the multiple simultaneous requests (e.g., col., 7).

13. Referring to claim 6, Parry discloses the claimed limitations as rejected above. Parry also discloses storage medium comprising a plurality of executable instructions including at least a subset of which that, when executed, implement a method according to claim 1 (e.g., col., 3).

14. Referring to claim 7, Parry discloses the claimed limitations as rejected above. Parry also discloses computing system comprising: a storage medium having stored thereon a plurality of executable instructions; and an execution unit, coupled to the storage medium, to execute at least

a subset of the plurality of executable instructions to implement a method according to claim 1 (e.g., col., 3).

15. Referring to claim 8, Parry discloses the claimed limitations as rejected above. Parry also discloses a method of generating a filter graph for a user- defined processing project (e.g., col., 3), the method comprising: analyzing the project for multiple accesses to a single source of media content (e.g., col., 6); determining that the multiple accesses cannot be combined and/or share a common processing chain (e.g., col., 6); and coupling a single instance of the media source to the one or more processing chains through a software object to satisfy the multiple accesses without invoking a commensurate number of multiple instances of the media source (e.g., col., 6), wherein the one or more processing chains comprise: a scalable, dynamically reconfigurable matrix switch having a plurality of inputs and a plurality of outputs (e.g., col., 7); at least one matrix switch input being communicatively linked with a first processing chain portion (e.g., col., 7); at least one other matrix switch input being communicatively linked with a second processing chain portion (e.g., col., 7); the matrix switch being configured to dynamically couple one or more of the matrix switch inputs to one or more of the matrix switch outputs (e.g., col., 7).

16. Referring to claim 9, Parry discloses the claimed limitations as rejected above. Parry also discloses receiving a request for content at the software object; and issuing a seek command from the software object to the media source to retrieve the media for presentation to a requesting processing chain (e.g., col., 7).

17. Referring to claim 10, Parry discloses the claimed limitations as rejected above. Parry also discloses wherein the method is implemented by a render engine, exposed from an operating system to a media processing system executing on a computing system (e.g., col., 6).

18. Referring to claim 12, Parry discloses the claimed limitations as rejected above. Parry also discloses method according to claim 8, further comprising: identifying multiple simultaneous access to the media source (e.g., col., 7); and invoking a commensurate number of software objects, coupling a commensurate number of instances of the media source to processing chains to satisfy the multiple simultaneous requests (e.g., col., 7).

19. Referring to claim 13, Parry discloses the claimed limitations as rejected above. Parry also discloses storage medium comprising a plurality of executable instructions including at least a subset of which that, when executed, implement a method according to claim 8 (e.g., col., 3).

20. Referring to claim 14, Parry discloses the claimed limitations as rejected above. Parry also discloses computing system comprising: a storage medium having stored thereon a plurality of executable instructions; and an execution unit, coupled to the storage medium, to execute at least a subset of the plurality of executable instructions to implement a method according to claim 8 (e.g., col., 3).

21. Referring to claim 15, Parry discloses the claimed limitations as rejected above. Parry also discloses method of generating a filter graph for a user- defined processing project (e.g., col., 3), the method comprising: analyzing the project for multiple accesses to a single source of media content (e.g., col., 6); determining that the multiple accesses cannot be combined and/or share a common processing chain (e.g., col., 6); coupling a single instance of the media source to the multiple processing chains through a software object to satisfy the multiple accesses without invoking a commensurate number of multiple instances of the media source (e.g., col., 6); and ascertaining whether the multiple processing chains share common pre- processing attributes and, if so, interposing one or more associated filters between the single source of media content and the software object (e.g., col., 6).

22. Referring to claim 16, Parry discloses the claimed limitations as rejected above. Parry also discloses receiving a request for content at the software object; and issuing a seek command from the software object to the media source to retrieve the media for presentation to a requesting processing chain (e.g., col., 7).

23. Referring to claim 17, Parry discloses the claimed limitations as rejected above. Parry also discloses wherein the method is implemented by a render engine, exposed from an operating system to a media processing system executing on a computing system (e.g., col., 3).

24. Referring to claim 19, Parry discloses the claimed limitations as rejected above. Parry also discloses identifying multiple simultaneous access to the media source; and invoking a

commensurate number of software objects, coupling a commensurate number of instances of the media source to processing chains to satisfy the multiple simultaneous requests (e.g., col., 7).

25. Referring to claim 20, Parry discloses the claimed limitations as rejected above. Parry also discloses storage medium comprising a plurality of executable instructions including at least a subset of which that, when executed, implement a method according to claim 15 (e.g., col., 3).

26. Claims 1-3, 5-10, 12-17, 19, 20 are rejected under 35 U.S.C. 102(e) as being anticipated by Lortz, 7047554 (Hereinafter Lortz).

27. Referring to claim 1, Lortz discloses a method of generating a filter graph for a user-defined processing project (e.g., col., 3), the method comprising: analyzing the project for multiple accesses to a single source of media content (e.g., col., 3); determining that the multiple accesses cannot be combined and/or share a common processing chain (e.g., col., 3); and coupling a single instance of the media source to the one or more processing chains through a software object to satisfy the multiple accesses without invoking a commensurate number of multiple instances of the media source (e.g., col., 3).

28. Referring to claim 2, Lortz discloses the claimed limitations as rejected above. Lortz also discloses receiving a request for content at the software object; and issuing a seek command from the software object to the media source to retrieve the media for presentation to a requesting processing chain (e.g., col., 4).

29. Referring to claim 3, Lortz discloses the claimed limitations as rejected above. Lortz also discloses wherein the method is implemented by a render engine, exposed from an operating system to a media processing system executing on a computing system (e.g., col., 3).

30. Referring to claim 5, Lortz discloses the claimed limitations as rejected above. Lortz also discloses identifying multiple simultaneous access to the media source; and invoking a commensurate number of software objects, coupling a commensurate number of instances of the media source to processing chains to satisfy the multiple simultaneous requests (e.g., col., 4).

31. Referring to claim 6, Lortz discloses the claimed limitations as rejected above. Lortz also discloses storage medium comprising a plurality of executable instructions including at least a subset of which that, when executed, implement a method according to claim 1 (e.g., col., 3).

32. Referring to claim 7, Lortz discloses the claimed limitations as rejected above. Lortz also discloses computing system comprising: a storage medium having stored thereon a plurality of executable instructions; and an execution unit, coupled to the storage medium, to execute at least a subset of the plurality of executable instructions to implement a method according to claim 1 (e.g., col., 3).

33. Referring to claim 8, Lortz discloses the claimed limitations as rejected above. Lortz also discloses a method of generating a filter graph for a user- defined processing project (e.g., col., 3), the method comprising: analyzing the project for multiple accesses to a single source of

media content (e.g., col., 3); determining that the multiple accesses cannot be combined and/or share a common processing chain (e.g., col., 3); and coupling a single instance of the media source to the one or more processing chains through a software object to satisfy the multiple accesses without invoking a commensurate number of multiple instances of the media source (e.g., col., 3), wherein the one or more processing chains comprise: a scalable, dynamically reconfigurable matrix switch having a plurality of inputs and a plurality of outputs (e.g., col., 4); at least one matrix switch input being communicatively linked with a first processing chain portion (e.g., col., 4); at least one other matrix switch input being communicatively linked with a second processing chain portion (e.g., col., 4); the matrix switch being configured to dynamically couple one or more of the matrix switch inputs to one or more of the matrix switch outputs (e.g., col., 4).

34. Referring to claim 9, Lortz discloses the claimed limitations as rejected above. Lortz also discloses receiving a request for content at the software object; and issuing a seek command from the software object to the media source to retrieve the media for presentation to a requesting processing chain (e.g., col., 4).

35. Referring to claim 10, Lortz discloses the claimed limitations as rejected above. Lortz also discloses wherein the method is implemented by a render engine, exposed from an operating system to a media processing system executing on a computing system (e.g., col., 3).

36. Referring to claim 12, Lortz discloses the claimed limitations as rejected above. Lortz also discloses method according to claim 8, further comprising: identifying multiple simultaneous access to the media source (e.g., col., 4); and invoking a commensurate number of software objects, coupling a commensurate number of instances of the media source to processing chains to satisfy the multiple simultaneous requests (e.g., col., 4).

37. Referring to claim 13, Lortz discloses the claimed limitations as rejected above. Lortz also discloses storage medium comprising a plurality of executable instructions including at least a subset of which that, when executed, implement a method according to claim 8 (e.g., col., 3).

38. Referring to claim 14, Lortz discloses the claimed limitations as rejected above. Lortz also discloses computing system comprising: a storage medium having stored thereon a plurality of executable instructions; and an execution unit, coupled to the storage medium, to execute at least a subset of the plurality of executable instructions to implement a method according to claim 8 (e.g., col., 3).

39. Referring to claim 15, Lortz discloses the claimed limitations as rejected above. Lortz also discloses method of generating a filter graph for a user- defined processing project (e.g., col., 3), the method comprising: analyzing the project for multiple accesses to a single source of media content (e.g., col., 3); determining that the multiple accesses cannot be combined and/or share a common processing chain (e.g., col., 3); coupling a single instance of the media source to the multiple processing chains through a software object to satisfy the multiple accesses without

invoking a commensurate number of multiple instances of the media source (e.g., col., 3); and ascertaining whether the multiple processing chains share common pre- processing attributes and, if so, interposing one or more associated filters between the single source of media content and the software object (e.g., col., 3).

40. Referring to claim 16, Lortz discloses the claimed limitations as rejected above. Lortz also discloses receiving a request for content at the software object; and issuing a seek command from the software object to the media source to retrieve the media for presentation to a requesting processing chain (e.g., col., 4).

41. Referring to claim 17, Lortz discloses the claimed limitations as rejected above. Lortz also discloses wherein the method is implemented by a render engine, exposed from an operating system to a media processing system executing on a computing system (e.g., col., 3).

42. Referring to claim 19, Lortz discloses the claimed limitations as rejected above. Lortz also discloses identifying multiple simultaneous access to the media source; and invoking a commensurate number of software objects, coupling a commensurate number of instances of the media source to processing chains to satisfy the multiple simultaneous requests (e.g., col., 4).

43. Referring to claim 20, Lortz discloses the claimed limitations as rejected above. Lortz also discloses storage medium comprising a plurality of executable instructions including at least a subset of which that, when executed, implement a method according to claim 15 (e.g., col., 3).

44. Claims 1-3, 5-10, 12-17, 19, 20 are rejected under 35 U.S.C. 102(a) as being anticipated by DeLeeuw, 6088018 (Hereinafter DeLeeuw).

45. Referring to claim 1, DeLeeuw discloses a method of generating a filter graph for a user-defined processing project (e.g., col., 4), the method comprising: analyzing the project for multiple accesses to a single source of media content (e.g., col., 4); determining that the multiple accesses cannot be combined and/or share a common processing chain (e.g., col., 4); and coupling a single instance of the media source to the one or more processing chains through a software object to satisfy the multiple accesses without invoking a commensurate number of multiple instances of the media source (e.g., col., 4).

46. Referring to claim 2, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses receiving a request for content at the software object; and issuing a seek command from the software object to the media source to retrieve the media for presentation to a requesting processing chain (e.g., col., 5).

47. Referring to claim 3, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses wherein the method is implemented by a render engine, exposed from an operating system to a media processing system executing on a computing system (e.g., col., 4).

48. Referring to claim 5, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses identifying multiple simultaneous access to the media source; and invoking a commensurate number of software objects, coupling a commensurate number of instances of the media source to processing chains to satisfy the multiple simultaneous requests (e.g., col., 5).

49. Referring to claim 6, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses storage medium comprising a plurality of executable instructions including at least a subset of which that, when executed, implement a method according to claim 1 (e.g., col., 4).

50. Referring to claim 7, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses computing system comprising: a storage medium having stored thereon a plurality of executable instructions; and an execution unit, coupled to the storage medium, to execute at least a subset of the plurality of executable instructions to implement a method according to claim 1 (e.g., col., 4).

51. Referring to claim 8, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses a method of generating a filter graph for a user- defined processing project (e.g., col., 4), the method comprising: analyzing the project for multiple accesses to a single source of media content (e.g., col., 4); determining that the multiple accesses cannot be combined and/or share a common processing chain (e.g., col., 4); and coupling a single instance

of the media source to the one or more processing chains through a software object to satisfy the multiple accesses without invoking a commensurate number of multiple instances of the media source (e.g., col., 4), wherein the one or more processing chains comprise: a scalable, dynamically reconfigurable matrix switch having a plurality of inputs and a plurality of outputs (e.g., col., 5); at least one matrix switch input being communicatively linked with a first processing chain portion (e.g., col., 5); at least one other matrix switch input being communicatively linked with a second processing chain portion (e.g., col., 5); the matrix switch being configured to dynamically couple one or more of the matrix switch inputs to one or more of the matrix switch outputs (e.g., col., 5).

52. Referring to claim 9, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses receiving a request for content at the software object; and issuing a seek command from the software object to the media source to retrieve the media for presentation to a requesting processing chain (e.g., col., 5).

53. Referring to claim 10, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses wherein the method is implemented by a render engine, exposed from an operating system to a media processing system executing on a computing system (e.g., col., 4).

54. Referring to claim 12, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses method according to claim 8, further comprising: identifying multiple

simultaneous access to the media source (e.g., col., 5); and invoking a commensurate number of software objects, coupling a commensurate number of instances of the media source to processing chains to satisfy the multiple simultaneous requests (e.g., col., 5).

55. Referring to claim 13, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses storage medium comprising a plurality of executable instructions including at least a subset of which that, when executed, implement a method according to claim 8 (e.g., col., 4).

56. Referring to claim 14, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses computing system comprising: a storage medium having stored thereon a plurality of executable instructions; and an execution unit, coupled to the storage medium, to execute at least a subset of the plurality of executable instructions to implement a method according to claim 8 (e.g., col., 4).

57. Referring to claim 15, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses method of generating a filter graph for a user- defined processing project (e.g., col., 4), the method comprising: analyzing the project for multiple accesses to a single source of media content (e.g., col., 4); determining that the multiple accesses cannot be combined and/or share a common processing chain (e.g., col., 4); coupling a single instance of the media source to the multiple processing chains through a software object to satisfy the multiple accesses without invoking a commensurate number of multiple instances of the media

source (e.g., col., 4); and ascertaining whether the multiple processing chains share common pre-processing attributes and, if so, interposing one or more associated filters between the single source of media content and the software object (e.g., col., 4).

58. Referring to claim 16, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses receiving a request for content at the software object; and issuing a seek command from the software object to the media source to retrieve the media for presentation to a requesting processing chain (e.g., col., 5).

59. Referring to claim 17, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses wherein the method is implemented by a render engine, exposed from an operating system to a media processing system executing on a computing system (e.g., col., 4).

60. Referring to claim 19, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses identifying multiple simultaneous access to the media source; and invoking a commensurate number of software objects, coupling a commensurate number of instances of the media source to processing chains to satisfy the multiple simultaneous requests (e.g., col., 5).

61. Referring to claim 20, DeLeeuw discloses the claimed limitations as rejected above. DeLeeuw also discloses storage medium comprising a plurality of executable instructions

including at least a subset of which that, when executed, implement a method according to claim 15 (e.g., col., 4).

Allowable Subject Matter

Claims 4, 11 and 18 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Conclusion

In order to expedite the prosecution of this case, multiple references are used for the rejections to demonstrate that several references disclose the claimed subject matter of the claims. Also please refer to the prior arts of the office actions of the copending applications, 09/731,490, 09/731529, 10/980,514, which are also part of the applicant submitted IDS(s).

Examiner has cited particular columns and line numbers and/or paragraphs and/or sections and/or page numbers in the reference(s) as applied to the claims above for the convenience of the applicant. Although the specified citations are representative of the teachings of the art and are applied to the specific limitations within the individual claim, other passages and figures may apply as well. It is respectfully requested from the applicant in preparing responses, to fully consider the references in entirety, as potentially teaching, all or part of the claimed invention, as well as the context of the passage, as taught by the prior art or disclosed by the Examiner.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Haresh Patel whose telephone number is (571) 272-3973. The examiner can normally be reached on Monday, Tuesday, Thursday and Friday from 10:00 am to 8:00 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Nathan Flynn, can be reached at (571) 272-1915. The fax phone number for the organization where this application or proceeding is assigned is (571) 273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Haresh N. Patel/

Primary Examiner, Art Unit 2154

6/20/2008